



# Nichts geht mehr

Es musste einfach sein. Viel zu lange hatte ich gezögert. Visual Studio 2022 musste endlich auf den Rechner.

**W**ebsite ansteuern. Download starten. Im Installer die gewünschten Komponenten und Entwicklungsbereiche auswählen, und los ging es mit der Installation. Sie dauerte überraschend kurz, obwohl die zu übertragende Datenmenge wie bei Visual Studio 2019 mehrere Gigabyte betrug. Ein erster Start nach der Installation bescherte ein Visual Studio mit dunklem Schema, was ich in etwas optimistisch Helles umschaltete. Projekte laden, ausführen, alles klappte. Noch ...

Der nächste Tag bescherte einen Ortswechsel aus dem Homeoffice ins Büro in der Münchner Innenstadt. Als ich dort meine selbstgeschriebene Artikelverwaltung öffnen wollte, gab es nur eine Fehlermeldung, die in etwa so lautete:

**Die .NET-6.0-irgendwas-DLL ist nicht zertifiziert.  
Deshalb wird deren Ausführung unterbunden.  
Mit freundlichen Grüßen, Ihr eifriger Virenschanner.**

Ich konnte aber unmöglich so lange warten, bis die Firma hinter dem Virenschanner die Dateien zertifiziert oder Microsoft .NET 6 final veröffentlicht hatte. Ich brauchte meine Artikelverwaltung.

Also kurzer Prozess: Visual Studio 2022 flog nach nur einem Tag wieder vom Rechner. Leider löste auch das mein Problem nicht. Im Gegenteil: Jetzt schien überhaupt keine .NET-Version mehr installiert zu sein. Aber glücklicherweise gibt es so hilfsbereite Menschen wie Hamid Mosalla. Hamid hatte einen Blogpost verfasst, der dieses Problem sehr systematisch angeht [1]. Gleich der erste Schritt seiner Anleitung zeigte das ganze Ausmaß des Desasters. Der Befehl

```
dotnet --list-sdks
```

listete keine einzige Version von .NET auf. Irgendwie hatte der Installer beim Deinstallieren zwar nicht die .NET-Versionen, so doch immerhin die Einträge im Suchpfad gelöscht. Nach dem Einfügen dieser Einträge funktionierte die Artikelverwaltung wieder. Trotzdem hinterlässt das Ganze einen schalen Geschmack: Immer noch entscheidet der Pfad, ob ein .NET-Programm läuft oder nicht. Fehlt nur noch, dass ich HIMEM.SYS oder EMM386.SYS laden muss. Wenn Ihnen diese beiden Namen nichts sagen, seien Sie froh: Sie sind verdammt jung.

Viel Spaß mit der dotnetpro

Tilman Börner  
Chefredakteur dotnetpro

[1] Unable to locate the .NET SDK: The Reasons, [www.dotnetpro.de/SL2112Edi1](http://www.dotnetpro.de/SL2112Edi1)



**Mike Bild**

stellt die neue Webarchitektur JAMStack vor (S. 8)



**Manuela Sayin**

geht der Frage nach, ob Code glücklich machen kann (S. 34)



**Christian Sender**

zeichnet mit dem 2D-Grafiksystem SkiaSharp (S. 41)